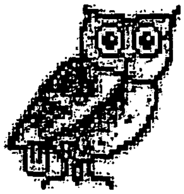
60/480,626

Product Requirements Document

VendCast Audit Device
Project Name: Glaux



Document Version:	**1.0**
Date:	**23-April-2002**
Author:	**Erin Defosse**

Document Control:

Version	Date	Author	Description
1.0a1	08-Apr-2002	E. Defosse	Initial Draft Release
1.0b1	09-Apr-2002	E. Defosse	Renamed document as a "Product Requirements Document" and added various details in the Detailed System Requirements section. Document is fairly complete now.
1.0b2	22-Apr-2002	E. Defosse	Made changes based on in-house review done on 18-Apr-2002. Major changes included: 1) removing requirement of having the AD know how to write Vender Asset Info to the VMC leaving it to the handheld to compose the required DEX and only requiring that the AD push the DEX to the VMC, 2) Divided the Install use case into two (one with a handheld, the other without a handheld), 3) Only required that AD (in Phase 1) work with cold drink venders, 4) Removed Vender Information data as explicit data fields and rolled it into the Ad-Hoc data.
1.0	23-Apr-2002	E. Defosse	Made small editorial changes to 1.0b2 and published as final doc (v 1.0). Added more detailed milestones and deliverable dates.

Approvals:

Who	Company	Signature	Date

1 Introduction

The purpose of this document is to detail the set of features required for the *VendCast Audit Device*, herein called *Glaux*. This project's goal is to develop an embedded data collection and storage device that, when placed in inside a vending machine, will collect both DEX and MDB data using a combination of ad-hoc scheduling and trigger based events. Data collected by Glaux will be subsequently transferred to a handheld computer which, in turn, will communicate the information to the VendCast host application.

2 Related Documents

The following documents contain information supporting this requirements document:

- **EVA-DTS:** *EVA-DTS, Release 5.0*, European Vending Association
- **MDB/ICP:** *MDB/ICP, Version 2.0*, NAMA, 4 October 2000
- **EVS:** *Electronic Vending Standard (EVS) 3.0;* The Coca-Cola Company
- **DEX/UCS:** *Uniform Communications Standards for Direct Store Delivery – Implementation and User Guide (UCS/DSD-IUG)*, Uniform Code Council

3 Project Overview

The purpose of the Audit Device, or Glaux, initiative is to provide a low cost alternative to the on-line collection of vender DEX and MDB data and to enable additional value-added capabilities at the vending machine. Glaux will integrate with a handheld software applications also being developed by Isochron in order to allow transport of the data collected by the Audit Device to a host application for further processing without requiring on-line connectivity between the host and the Audit Device.

4 Market Requirements

4.1 Low Cost

Provide a low cost alternative to the on-line collection of data from a vending machine

4.1.1 Cheap enough to seriously consider it for mass deployment in a large (>20%) fraction of the current vender installed base

4.1.2 Elimination of recurring costs normally associated with on-line solutions (e.g. wireless communications costs.

4.2 Rich Data

Provide a rich data set capable to enable the generation of meaningful and useful information by the VendCast host application

4.2.1 Route Driver Dispatch

4.2.2 Space to Sales Optimization

4.2.3 Cash Accountability

4.2.4 Vender Lifecycle Management

4.2.5 Audit Device Management

4.3 Ease of Use

Be easy for a route driver or service technician to work with

4.3.1 No special buttons to press

4.3.2 "Connect Once" at the vender and leave

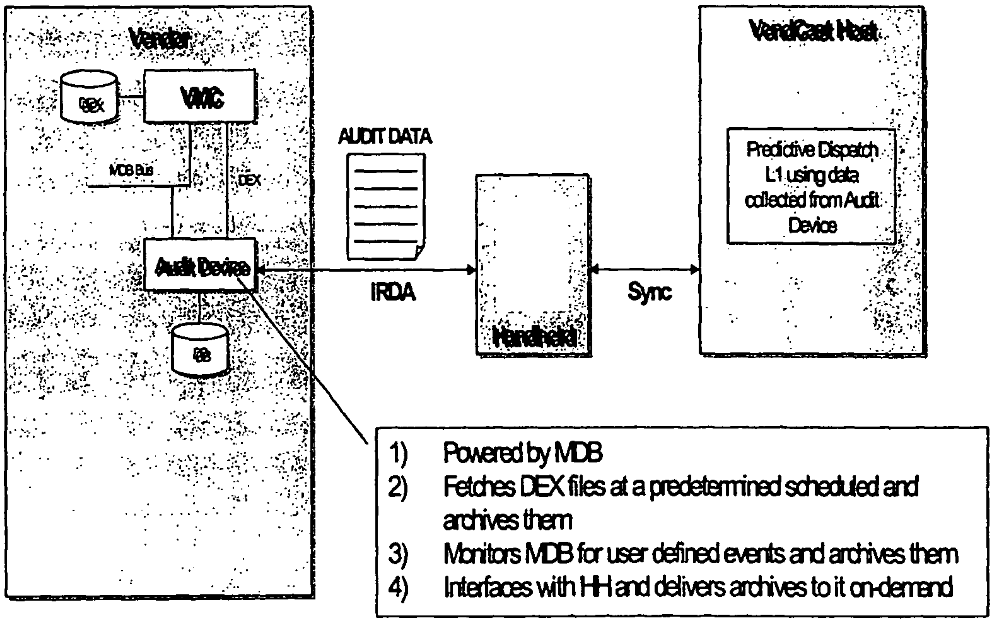
4.3.3 Provide meaningful debug information to service technicians

4.3.4 The solution must work in conjunction with commonly used handheld computers.

5 Response to Market Requirements

5.1 Solution Architecture

Figure 6.1.1 illustrates the solution architecture selected for addressing the Market Requirements. The solution consists of placing an embedded processor (Audit Device) inside the vending machine operable to obtain DEX data from the VMC and MDB data being generated by the VMC and the payment peripherals on-board the vender. The data is archived in non-volatile memory and, when a handheld with the appropriate software interfaces with the Audit Device, these archives, in addition to the then current DEX and MDB peripheral data, is transferred on-demand to the handheld using an IRDA interface. The handheld is subsequently interfaced with the VendCast host application and data downloaded to and from it using any number of syncing or data transfer mechanisms. The VendCast host will use the collected data for, amongst other things, enabling route dispatch using predictive techniques.



1) Powered by MDB
2) Fetches DEX files at a predetermined scheduled and archives them
3) Monitors MDB for user defined events and archives them
4) Interfaces with HH and delivers archives to it on-demand

Figure 6.1.1: Solution Architecture

Key benefits of the solution architecture as described herein include:

5.1.1 Elimination of monthly wireless communication fees by using the handheld to shuttle the data from the vender location to the VendCast host application.

5.1.2 Make the unit low cost by eliminating expensive radios, highly tuned power supplies, multiple I/Os, RAM, etc. currently present in our existing hardware solution

5.1.3 Make it compatible with off-the-shelf handheld devices (i.e. do not require expensive add-ons) by using IRDA and/or cable connection with handheld.

5.1.4 Software on-board the handheld for communicating with the Audit Device will be integrated with the VendCast handheld software in order to provide the user a single application that takes care of all his/her needs. The handheld implements all of the workflow associated with the use of the solution.

5.1.5 Makes the Audit Device the single interface point for all data transactions at the vender by using the Audit Device handheld interface as the gateway for all communications (including programming of the VMC). This makes the system easy to use for the customer by reducing interface points and eliminating the issues associated with wired interfaces.

5.2 Hardware Architecture

In order to implement the solution architecture as presented herein, the Audit Device must support a variety of interfaces and internal subsystems. A diagram illustrating the hardware elements necessary for the implementation of these interfaces and subsystems is presented in **Figure 5.2-1**

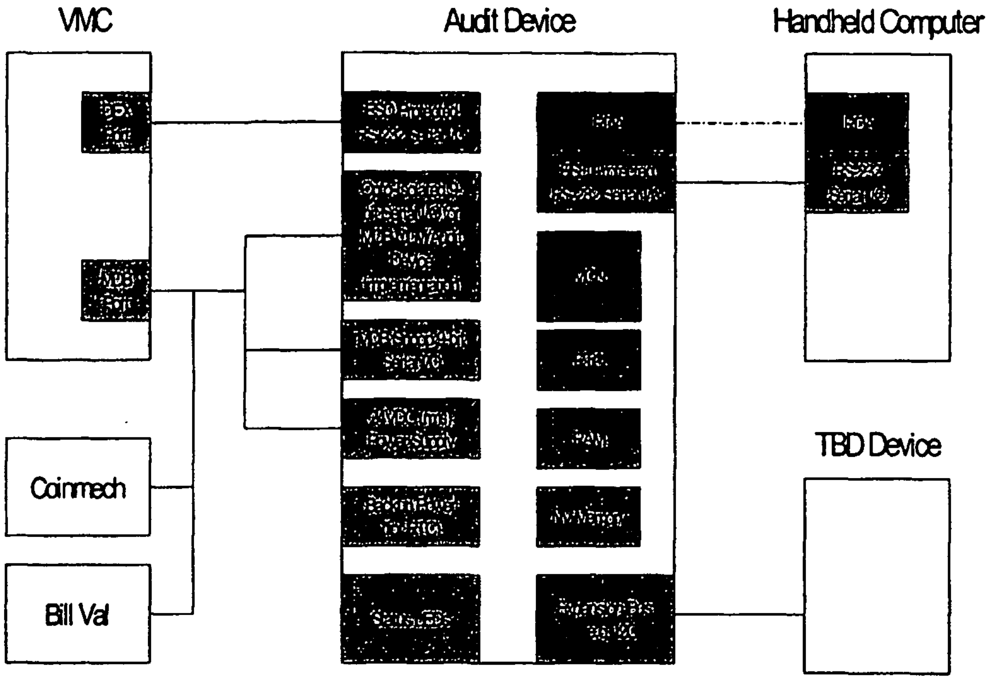


Figure 5.2-1: Hardware Architecture (Phase 1)

5.2.1 DEX Interface

The DEX Interface is an RS-232 serial data I/O interface that will be used to obtain DEX data from the vender's VMC. The interface must provide ESD protection as well as deal with the various real-world implementations of DEX in vending machines. For example, the interface must be able handle Master, Slave, and Slave-Read Only VMC's in software and, in hardware, be able to go in and out of a very high-impedance mode so that VMC's that attempt to detect devices connected on their VMC port cannot detect the Audit Device when it is not collecting data from the VMC.

5.2.2 MDB Interface

The MDB Interface is a serial data I/O interface that will be used to obtain atomic level sales transaction data as well as payment peripheral status data (e.g. error codes, aborted sales transactions, etc.). The MDB interface will be designed to operate in three modes: MDB Snoop, OLM, and MDB Audit Device.

The MDB interface is implemented in hardware using an opto-isolated circuit design or an equivalent design that provides for very high transient protection to the Audit Device. In addition, reading and writing data over the MDB will require the Audit Device to implement 9-bit serial data which is non-standard.

5.2.3 Handheld Interface

The Handheld Interface provides a mechanism for getting and setting data to and from the Audit Device using a handheld computer. From a software perspective the Handheld Interface should be implemented as a Master-Slave where the handheld is the master and the Audit Device is a slave. A simple command-response protocol should be implemented to allow the handheld to get and set data to and from the Audit Device. For purposes of file transfer between the handheld and the Audit Device, a standard file transfer protocol such as Zmodem should be implemented.

5.2.4 Expansion Bus

The Expansion Bus is intended to provide a means for the Audit Device to interface directly with other hardware that may eventually be available inside the vender. For example, an Isochron-developed Cashless Unit that accepts a variety of payment tokens. In this case, it is desirable to re-use the handheld interface on either of the devices so that a user can interface once and perform all transactions necessary. The bus should provide bi-directional data (e.g. I2C) as well as power input (to drive the Audit Device) and power output (to drive an external device).

5.2.5 Power Subsystem

The Power Subsystem is responsible for providing operational power to all electronics on the Audit Device. Power to drive the operation of the unit will be obtained via the MDB interface and, as such, appropriate power regulation and surge suppression must be provided in order to deal with the transient fluctuations that occur on MDB.

5.2.6 Data Storage Subsystem

The Data Storage Subsystem provides for non-volatile storage for data obtained via the DEX, MDB, and Handheld interfaces. For example, DEX files, MDB peripheral status, Audit Device configuration, POC data, etc.

5.2.7 Timing Subsystem

The Audit Device must be able to maintain a RTC synchronized to a reference standard so that it can timestamp archived data. As such, backup power must be provided in a way that allows the Audit Device to maintain its clock even during extended vender power down conditions.

5.2.8 Processor Subsystem

The Processor Subsystem consists of the embedded microprocessor and associated RAM necessary to drive the software and hardware functionality on the Audit Device.

5.2.9 User Interface Subsystem

The User Interface Subsystem on the Audit Device will consist exclusively of LED indicators on the unit which will provide core operational status feedback to the user.

6 Software Architecture

The Audit Device must implement the necessary software functionality in order to meet the Market Requirements. The requirements are primarily data driven, that is, that collection, maintenance, and delivery of data is the primary objective of the Audit Device. As such, the software architecture has been developed using an Information Architecture as its source model. Specific functionality required for the Audit Device is derived by examining the various Use Cases of the system. The Information Architecture as well as key Use Cases are presented below.

6.1 Information Architecture

The Audit Device is part of data driven solution for our customers. The primary objective of the Audit Device is to collect, maintain, archive, and deliver data regarding the operation of a vender over time. The specific types of data objects necessary to meet the Market Requirements are:

6.1.1 DEX Audit Data

DEX Audit data consists of the combination of archived DEX audit objects and the most current DEX audit object. A DEX audit object consists of the a) the DEX file obtained from the VMC, b) the GMT timestamp associated with the DEX file (the date and time at which the file was obtained from the VMC), and c) a Status field indicating the condition of the DEX Interface at the time that the audit attempt was made.

The status field shall indicate if the DEX Interface was in a "normal" state (meaning that the Audit Device is capable of retrieving DEX files) or if it is a "not communicating with VMC" state (meaning that the Audit Device is unable to communicate with the VMC and retrieve a DEX file). Additionally, if possible, a "not communicating with VMC" status should be expanded to include the exact reason behind the communications failure. Examples of such failures include a) bad DEX session password, b) timing failure at the protocol level, c) no response from VMC after initial session inquiry (0x05), or d) some other protocol level failure.

6.1.1.1 Current DEX

A Current DEX object is generated on demand by a request from the handheld computer and, as such, represents the most current DEX audit object available at the time that the request was generated by the handheld. Every time a Current DEX is requested and delivered to the handheld it will be added to the Archived DEX collection.

6.1.1.2 Archived DEX

Archived DEX consists of a collection of DEX audit objects which have been collected over a period of time by the Audit Device, including any Current DEX objects collected on demand by the Audit Device. The objects which make up archived DEX are collected based on a predetermined schedule programmed on the Audit Device. The schedule can consist of any combination of collection frequencies (e.g. every day at 20:00 hours) or any number of ad-hoc schedules (e.g. Monday at 12:00 and Thursday at 14:00).

The Archived DEX data shall be stored on the Audit Device is non-volatile memory in a compressed format. Ideally, a ZLIB with Dictionary compression algorithm [NOTE: this type of compression is described in another patent filed by Isochron] should be used in order to minimize the amount of non-volatile memory needed to store the archive. This data is to be delivered on-demand to the handheld.

Once the handheld downloads records from the archive these records shall be marked as "delivered" and will not be sent to the handheld on subsequent downloads unless specifically requested by the handheld. In the event that the size of the archive exceeds the physical memory allocation provided for it, the archive will act as a FIFO buffer whereby the oldest (based on timestamp) records are removed in order to make room for new ones.

6.1.2 MDB Audit Data

MDB Audit Data consists of a variety of different information elements which can be obtained by either listening in on the MDB using MDB Snoop. In MDB Snoop, the Audit Device will listen to communications being carried on both MDB data lines (VMC transmit and VMC receive) and examining the contents of the data being communicated between the VMC and MDB peripherals.

6.1.2.1 Current Peripheral Status

Current peripheral status refers to the operational status of the peripherals currently installed on the vender's MDB. Examples of peripherals include bill validators, coin mechanisms, card readers, etc. The status of each of these peripherals can be determined by using MDB Snoop mode examining the history of the responses by the peripherals to the Poll command routinely sent to them by the VMC on the bus. Typically a peripheral will respond with a "normal" status response but, in the event of a problem with the peripheral, will respond with one of many different error codes documented in the MDB/ICP spec. Furthermore, it is important to note that peripherals are required to only transmit these detailed error codes once immediately after the error occurs, with subsequent responses to the STATUS command being a generic "peripheral disabled" response. Therefore, it is necessary for the Audit Device to track not only the current response to the Poll command but to also record the detailed error code and GMT timestamp associated with the original error event.

6.1.2.2 Peripheral Status History

Peripheral Status History is an archive of all MDB peripheral status events that have been recorded by listening in on the MDB. This archive consists of a series of GMT timestamped status change events. By timestamp and recording the peripheral ID (e.g. the type of peripheral as defined in the MDB/ICP spec) and every change in the device's response to the VMC's Poll command it is possible to construct a complete history of the events that have occurred. This history must be archived in non-volatile memory and transmitted on-demand to the handheld computer. The handheld computer can then analyze the history as needed.

6.1.2.3 Sales Transaction History

The Sales Transaction History consists of a series of GMT timestamped sales transactions which represent the complete set of successful and aborted vends at the vender. These events can be tracked through MDB Snoop.

The Sales Transaction History will be transmitted on demand to the handheld computer.

6.1.2.3.1 Successful Vends

The Audit Device will be able to monitor transactions on the MDB to determine when a successful vend has occurred. The

Audit Device will store the timestamp, sales value, and identify the peripheral(s) that provided the credit.

6.1.2.3.2 Exact Change Aborted Vends

Exact Change Aborted Vends are those vends that are aborted by the VMC due to an exact change condition at the vender. That is, a bill validator or coin transaction which was rejected by the VMC because there was no change available in the vender to proceed with the vend. This event can be identified, for example, by noting when the bill validator places a cash escrow value on the VMC in response to the VMC's Poll command and the VMC immediately orders the bill validator to return the bill using the Escrow command.

Once the handheld downloads records from the archive these records shall be marked as "delivered" and will not be sent to the handheld on subsequent downloads unless specifically requested by the handheld. In the event that the size of the archive exceeds the physical memory allocation provided for it, the archive will act as a FIFO buffer whereby the oldest (based on timestamp) records are removed in order to make room for new ones.

6.1.3 Time

The Audit Device will maintain an internal clock that is synchronized to GMT time. The Time on the Audit Device will be synchronized with GMT time available on the handheld computer every time that the handheld interfaces with the Audit Device. The handheld will be able to read the time on the Audit Device on-demand in order to validate it.

This Time will be used to timestamp all recorded events such as DEX Audit objects, MDB audit events, etc.

6.1.4 Audit Device Information

6.1.4.1 Asset Data

The Audit Device will maintain the asset data necessary in order to support the field management of the unit. This data will consist of the Audit Device serial number, model number, hardware revision number, manufacture date, and firmware revision number and date.

6.1.4.2 Configuration Data

The Audit Device will maintain a set of internal configuration parameters necessary for the operation of the device. These configuration parameters will include the DEX audit schedule, detailed DEX Interface parameters (e.g. packet timeout, character timeout, VMC type detection mode, VMC type list, etc.), an MDB audit event trigger table, etc. Detailed information on the various configuration parameters is found later in the detailed requirements section of this document.

6.1.4.3 General Events

The Audit Device will timestamp and general events which occur at the unit. General Events include power on/off events, firmware upgrade events, etc. Detailed information on the various events that must be logged can be found later in the detailed requirements section of this document. Once the handheld downloads records from the log these records shall be marked as "delivered" and will not be sent to the handheld on subsequent downloads unless specifically requested by the handheld. In the event that the size of the log exceeds the physical memory allocation provided for it, the archive will act as a FIFO buffer whereby the oldest (based on timestamp) records are removed in order to make room for new ones.

6.1.4.4 Handheld Transaction Log

The Audit Device will maintain a log of all transactions conducted between it and the handheld. Once the handheld downloads records from the log these records shall be marked as "delivered" and will not be sent to the handheld on subsequent downloads unless specifically requested by the handheld. In the event that the size of the archive exceeds the physical memory allocation provided for it, the archive will act as a FIFO buffer whereby the oldest (based on timestamp) records are removed in order to make room for new ones.

6.1.5 Ad-Hoc Data

The Audit Device will provide non-volatile memory storage space so that the handheld can write and read ad-hoc data as required. It will be up to the handheld to manage the storage space associated with the Ad-Hoc Data. The Ad-Hoc Data storage space may be used, amongst other things, to store a data structure (e.g. XML) containing information such as the vending machine asset information, space-to-sales information, selection and sku information, etc.

6.1.6 Ad-Hoc DEX Write

The Audit Device will make volatile (e.g. RAM) memory available in order to enable the handheld to write ad-hoc DEX strings to the VMC. The handheld application will compose a DEX write string that contains commands to be uploaded to the VMC as defined by the appropriate DEX specifications. It will be the responsibility of the Audit Device to receive that DEX write string, packetize it as required by the DEX specification, issue a write password to the VMC, and upload the write string. Any response received from the VMC which appears in the form of a DEX string will be passed back to the handheld.

The data objects and the flow of information into and out of the Audit Device are depicted graphically in the information architecture diagram presented in **Figure 6.1-1**. The diagram specifically notes which elements of the solution are responsible for GET and SET operations related to the data.

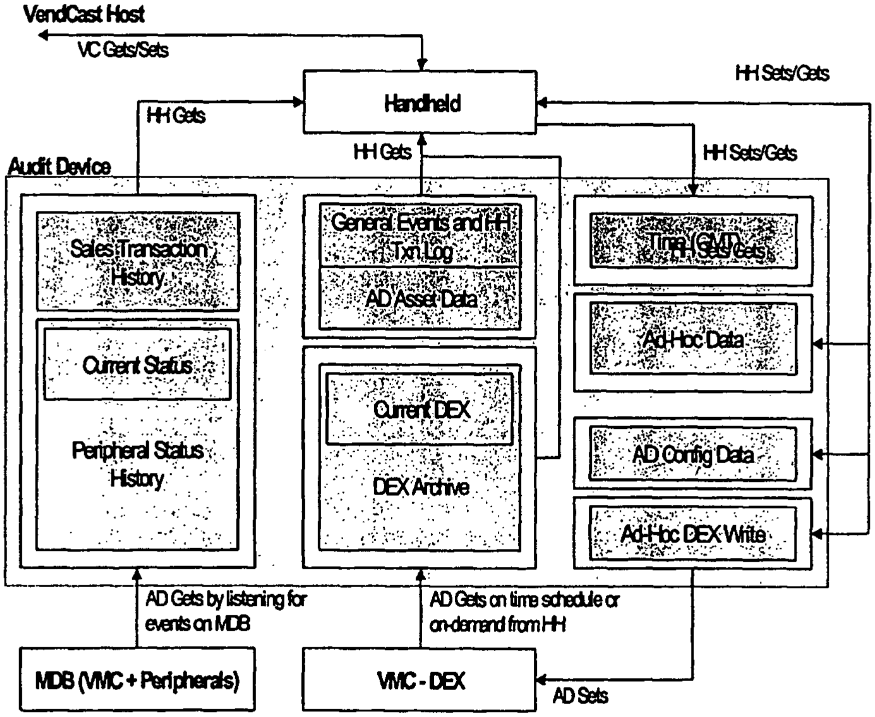


Figure 6.1-1: Information Architecture

6.2 Use Cases

The following Use Cases depict the typical ways in which the Audit Device will be used. The Use Cases identify various implementation requirements associated with the Audit Device. These requirements are defined in the Detailed System Requirements section of this document.

6.2.1 Installation without Handheld

Installation of the Audit Device on the vender may occur without the aid of a handheld device. Therefore, the Audit Device must be able to present sufficient status information via its LEDs to indicate to the installer that it has been correctly installed. Furthermore, it is expected that the Audit Device will adopt a "default" operating mode when installed without a handheld. One thing that must be considered is that, without a handheld, it is impossible to guarantee that the device has valid configuration parameters (e.g. time) and therefore interfacing with a handheld subsequent to this installation mode is highly recommended.

1) The Audit Device is physically mounted inside the vending machine
    a. Power to the vender door is shut off or power to the entire vender is removed.
    b. The Audit Device's DEX harness (phono plug) is inserted into the DEX port for the VMC
    c. The Audit Device's MDB harness is attached to the MDB.
    d. Power is restored to the vender door (or vender itself). This will cause power to flow through the MDB and turn on the Audit Device.
    e. At power up the Audit Device will first perform a self-diagnostic. If the self-diagnostic fails (e.g. a memory parity error) then the Audit Device will light its LEDs in a distinct fashion to alert the user of the problem. If a problem of this nature is encountered the install will not be able to proceed without a handheld device. Suggested alternative is to try using another Audit Device.
    f. After the unit is successfully booted the Audit Device will establish communications over the DEX and MDB Interfaces.
        i. Audit Device sets up the DEX Interface as necessary to manage DEX password, VMC Type detection, etc.
        ii. Audit Device sets up MDB Interface as necessary to snoop on the MDB. In future releases, this set up will include registering the device on the MDB as either an MDB OLM device or an MDB Audit Device per the MDB/ICP specification and/or the EVS specification.
        iii. LEDs on the Audit Device provide visual feedback on the success of the DEX and MDB interface setup. If a problem is detected, the user can attempt to correct the issue through manipulation of the DEX and MDB configuration parameters on the Audit Device with the handheld.
2) Set Audit Device to default operating mode

a. Installer uses a special tool (small screwdriver, etc.) to depress the "RESET TO FACTORY DEFAULTS" button located on the device. This performs a SOFT reset and clears all data stored in non-volatile memory on the device with the exception of the reference Time. By depressing the button during boot up the user can initiate a HARD reset which will clear all data stored in non-volatile memory, reset the clock, and clear the reference time.

b. After pressing the button the unit will return to a default operating mode.

    i. DEX Audit schedule is set to 1 poll per day at 00:00 hours GMT. If the internal clock is not synchronized then the poll occurs every 24 hours with boot-up or a hard reset marking zero time.

    ii. MDB audit of peripheral status is performed using the default trigger table (to be defined in a later phase of the Glaux project).

c. The unit will go through the same interface setup as described in step 1f above.

6.2.2 Installation with Handheld

Installation of the Audit Device on the vender may occur in the field or in the bottler's cooler shop. This task is expected to be performed by a person with technical skills and trained in the installation of the Audit Device. This installation process REQUIRES that a handheld with software operable to configure the Audit Device be present in order to perform a complete installation. If a handheld is not available the Audit Device will, after boot-up, default to a basic operating state (see Installation without Handheld use case).

3) The Audit Device is physically mounted inside the vending machine

    a. Power to the vender door is shut off or power to the entire vender is removed.

    b. The Audit Device's DEX harness (phono plug) is inserted into the DEX port for the VMC

    c. The Audit Device's MDB harness is attached to the MDB.

    d. Power is restored to the vender door (or vender itself). This will cause power to flow through the MDB and turn on the Audit Device.

    e. At power up the Audit Device will first perform a self-diagnostic. If the self-diagnostic fails (e.g. a memory parity error) then the Audit Device will light its LEDs in a distinct fashion to alert the user of the problem. If a problem is encountered the user may attempt to diagnose the issue with a handheld (e.g. download new firmware, soft reset the unit, hard reset the unit, etc.).

f. After the unit is successfully booted the Audit Device will establish communications over the DEX and MDB Interfaces.

i. Audit Device sets up the DEX Interface as necessary to manage DEX password, VMC Type detection, etc.

ii. Audit Device sets up MDB Interface as necessary to snoop on the MDB. In future releases, this set up will include registering the device on the MDB as either an MDB OLM device or an MDB Audit Device per the MDB/ICP specification and/or the EVS specification.

iii. LEDs on the Audit Device provide visual feedback on the success of the DEX and MDB interface setup. If a problem is detected, the user can attempt to correct the issue through manipulation of the DEX and MDB configuration parameters on the Audit Device with the handheld.

4) Service Tech uses the handheld to set up the basic configuration parameters of the Audit Device. For this purpose, the Service Tech takes the handheld and points its IRDA window to the IRDA window of the Audit Device.

a. Initial Handshake

i. Audit Device and handheld handshake in order to discover each other. The handshake process includes authentication that the handheld is an authorized handheld. If the authentication fails, the handheld is denied access to the Audit Device over the IRDA interface.

ii. After the handheld is authenticated, the Audit Device proceeds to synchronize its internal clock with the clock on the handheld. This routine synchronization provides the Audit Device with a reliable reference standard for which to run its RTC off of.

a. Service tech then accesses the appropriate screen on the handheld application and programs a variety of basic configuration parameters and POC parameters on to the Audit Device.

i. Configure the DEX audit data collection schedule and update the MDB trigger table

ii. Set Vender Asset, Selection, and S2S data in the Audit Device's non-volatile memory. On command from the handheld the Audit Device will write this data to the VMC.

iii. Store Ad-Hoc Data as required

b. Instead of manually setting all configuration parameters, the tech can access predefined configuration "profiles" in the handheld software and apply those to the Audit Device. This saves the tech time and reduces the possibility of errors.

c. The service tech may attempt to validate the configuration of the VMC after programming of the Audit Device is complete. For this purpose, s/he commands the Audit Device to obtain a DEX snapshot from the VMC. The information from this DEX snapshot is passed back to the handheld for inspection.

5) The Audit Device logs all handheld activity in the Handheld Transaction Log
6) Installer may choose to depress the "RESET TO FACTORY DEFAULTS" button at any time after the unit has powered on in order to initiate a soft-reset and re-start the setup process. The installer may also choose to perform a HARD reset by depressing the button during bootup.

6.2.3 Audit (Steady State)

The Audit use case represents the steady state operation of the Audit Device once it has been installed, configured, and placed in the field with a live vender. The Audit Device will perform automatic collection of DEX and MDB audit data as defined by the DEX audit schedule information and the MDB event triggers programmed into it during the installation phase (or as defined by its default settings).

1) AD follows pre-defined data collection configuration
    a. DEX: at a given frequency (Daily, Weekly, etc.) at a certain time-of-day or on specified days of the week and at a specified time-of-day
    b. MDB: based on user defined triggers such as a change in peripheral status, successful sales transactions, aborted sales transactions.
    c. As data is collected it is timestamped and stored/archived into non-volatile memory
2) If a power outage occurs, the Audit Device must maintain its RTC for up to 3 months. In the event that the outage is longer then, when the unit comes back up, it will begin timestamping using time from boot-up. The handheld software can later attempt to deduce actual date/time – note: this only works for the data acquired after the last power cycle.
3) The data collected during the audit phase corresponds to the data elements defined in the Information Architecture section of this document.

6.2.4 Driver Visit

The route driver will visit venders on some frequency (variable or fixed) in order to deliver product and collect cash. Every time the driver visits a vender s/he will, in addition to delivering product and collecting cash, use their route handheld to download audit information from the Audit Device and upload any data needed to update the operations of the Audit Device or vender itself. At the end of the day, the route driver will interface their handheld to the VendCast host application and the information collected from all vender's visited will be downloaded from the handheld to the host application. A route driver visit to a vender will consist of the following steps:

1) For the purpose of downloading the latest Audit Data on to his or her handheld, the route driver opens door and points the handheld's IRDA transceiver window to IRDA transceiver window on the Audit Device. When the Audit Device and the handheld have established (physical) IRDA communications an LED on the Audit Device turns on to indicate that this is the case (the LED will turn off if IRDA connectivity is lost). A similar indicator is provided in software on the handheld for the same purpose.

   a. Initial Handshake
      i. Audit Device and handheld handshake in order to discover each other. The handshake process includes authentication that the handheld is an authorized handheld. If the authentication fails, the handheld is denied access to the Audit Device over the IRDA interface.
      ii. After the handheld is authenticated, the Audit Device proceeds to synchronize its internal clock with the clock on the handheld. This routine synchronization provides the Audit Device with a reliable reference standard for which to run its RTC off of.
   b. The handheld prompts the driver on the possible actions that can be taken at this point. The driver selects to download all audit data from the Audit Device to the handheld. The process of selecting an action should tie into the Atlantis software workflow.
      i. The handheld software commands the Audit Device to deliver all archived audit data as well as the most current DEX and MDB peripheral status data. To get the most current DEX, the Audit Device actively polls the VMC for a VMC, creates a Current DEX audit object and prepares it for transfer to the handheld. The handheld application is responsible for performing "refill" DEX tags, if necessary.
      ii. The data is transferred over the IRDA interface and a progress bar provides feedback to the user regarding the status of the download. Nominally, transferring all audit data should not take more than 10 seconds. Once transfer is complete the user is notified.
      iii. The handheld analyzes the MDB audit data and alerts the user if it has determined that there is a problem with one of the MDB peripherals. If there is a problem, the driver may attempt to correct it. Once the attempt to correct is completed, the driver can use the handheld to re-request the most recent MDB peripheral status to determine if the corrective actions have been successful.

        iv. The handheld performs a consistency check on the assets by examining the Vender Information as well as the Audit Device Asset data downloaded from the Audit.
            1. If the Vender Information stored in non-volatile memory on the Audit Device does not match the information that appears in DEX then the handheld prompts the user to take corrective action. If no corrective action is possible, the audit data downloaded is marked as suspect, stored, and subsequently delivered to the host.
            2. If the Audit Device Asset data does not match the internal records stored within the handheld (e.g. the Audit Device – Vender association in the VendCast host application does not match what is found locally) then the handheld prompts the user to take corrective action.
        v. The Audit Device preserves all of the audit data after it has been downloaded to the handheld. The downloaded data is marked as "delivered" but not deleted from non-volatile storage. As the size of the DEX and MDB audit archives increases the oldest data is overwritten with newer data (FIFO). During subsequent driver visits only non-delivered data is downloaded, unless the driver specifically requests that all data be downloaded. By only downloading the most recent data we can minimize the file transfer time from the Audit Device to the handheld. However, the handheld may request records regardless of their "delivered" status using an ad-hoc record query mechanism.

2) In addition to downloading the audit data, the driver may want to make changes to the configuration of the Audit Device or the vender. The types of actions which are conceived of at this point are:
    a. Configure the DEX audit data collection schedule and update the MDB trigger table
    b. Set Vender Asset, Selection, and S2S data in the Audit Device's non-volatile memory.
    c. Store Ad-Hoc Data: this is a catchall to enable future capabilities.
    d. Soft reboot the Audit Device in an attempt to resolve a serial communications problem, for example.
3) The Audit Device logs all handheld activity in the Handheld Transaction Log

6.2.5 Service Tech Visit

A Service Tech will visit a vender from time to time as service problems are reported by customers, route drivers, of the VendCast host application itself. The Service Tech will arrive at the vender and, based on the type of problem reported, will attempt to diagnose the problems at the vender. The Service Tech will have with him/her a handheld device which will allow him/her to communicate with the Audit Device for the purpose of allowing him/her to inspect the operational status of the MDB peripherals, the VMC, and the Audit Device itself and will attempt to correct any problems encountered. In some other cases, the Service Tech may be called upon to modify the configuration of the vending machine to, for example, correct the selection to column mapping on the machine.

1) If there is a problem with the Audit Device the tech will first inspect the status of the LEDs. If any of the LEDs is displaying a problem condition then the tech will take appropriate action.
    a. If LEDs indicate that no power is present then the tech will attempt to reinstall the cabling or check the overall power condition of the vender.
    b. If LEDs indicate an interface problem (DEX or MDB) then the driver may choose to simply reboot the unit by power cycling it, resetting all cables, or proceed to use the handheld to diagnose the problem (see below).
2) For the purpose of inspecting the operational status of the Audit Device, the MDB peripherals, or the VMC itself, the tech points the handheld's IRDA transceiver window to IRDA transceiver window on the Audit Device.
    a. Initial Handshake
        i. Audit Device and handheld handshake in order to discover each other. The handshake process includes authentication that the handheld is an authorized handheld. If the authentication fails, the handheld is denied access to the Audit Device over the IRDA interface.
        ii. After the handheld is authenticated, the Audit Device proceeds to synchronize its internal clock with the clock on the handheld. This routine synchronization provides the Audit Device with a reliable reference standard for which to run its RTC off of.
        iii. If the handshake fails (for non security related reasons) this may indicate a problem with the Audit Device
    b. Problem Assessment: the handheld prompts the service tech on the possible actions that can be taken at this point. The service tech selects to view the operational status of the VMC, the MDB peripherals, and the Audit Device itself. The process of selecting an action should tie into the Atlantis software workflow.
        i. The handheld software then proceeds to download all MDB audit data, General Events, Vender Information, and Audit

Device Asset Data, and commands the Audit Device to obtain a DEX file from the VMC. The data downloaded is NOT marked as "delivered" in order to not interfere with the normal audit data collection process performed by route drivers.

    ii. The MDB audit data is analyzed to determine the status of the peripherals and their status history.
    iii. The General Events are analyzed to determine if any internal error conditions have been logged and to determine the power cycle history of the vender/Audit Device.
    iv. The DEX file is examined and its configurations (e.g. selection info, space-to-sales) compared with those stored in the Vender Information data objects. If the Audit Device is unable to download a DEX file then this is also noted.
    v. After the analysis of the data is complete the tech is presented with the results of the analysis. The tech can view the summary results or drill down on the history of, for example, an MDB peripheral's status over time.
  c. At this point the tech can take action, as required, to correct any problems detected during the analysis of the data.
    i. Configure the Audit Device and/or the VMC (data collection schedules, DEX and MDB interface parameters, VMC programming, etc.). Instead of manually setting all configuration parameters, the tech can access predefined configuration "profiles" in the handheld software and apply those to the Audit Device. This saves the tech time and reduces the possibility of errors.
    ii. Soft or hard reboot the Audit Device
    iii. Upload new firmware to the Audit Device

3) Other functions that may be carried out by the tech include:
  a. View Handheld Transaction Log
  b. Replace the Audit Device with a new one and perform the activities associated with the Install use case.
4) Audit Device logs all activity in the Handheld Transaction Log

6.2.6 Interface with Cashless Unit

This Use Case for the Audit Device involves a situation where the Audit Device is installed inside a vender together with an Isochron Cashless Unit. The Cashless Unit is a piece of hardware designed to enable the acceptance of non-cash payment tokens at the vending machine (e.g. credit card, Speedpass RFID, etc.). It is desired that, in this scenario, the single point of interface at the vending machine that was presented in the previous Use Cases, where the driver or tech had only to interface with one device, be maintained. In this setup, the Audit Device and the Cashless Unit are physically interfaced using the Expansion Bus on the Audit Device. A similar bus is assumed to exist on the Cashless Unit. The IRDA interface on the Audit Device would provide access to data on the Cashless Unit and would allow the handheld to execute commands on the Cashless Unit.

The physical setup of this Use Case would require that the Audit Device and Cashless Unit be connected using a cable harness or mounted together at their expansion bus points. Both units would maintain their respective MDB connections.

6.3 Handheld and Audit Device Interaction

6.3.1 Master-Slave Relationship

Based on the Information Architecture and Use Cases presented, it is desirable that the handheld and Audit Device interact using a Master-Slave relationship where the handheld drives all workflow related functionality and the Audit Device simply exposes various methods and properties (data) to the handheld. This design approach allows for maximum flexibility in the system and makes the system more scalable given that modifications to the functionality can be made by upgrading the code on the handheld rather than requiring that every Audit Device be upgraded.

Under this type of architecture, a typical interaction between a handheld and the Audit Device is depicted in **Figure 6.3.1-1**

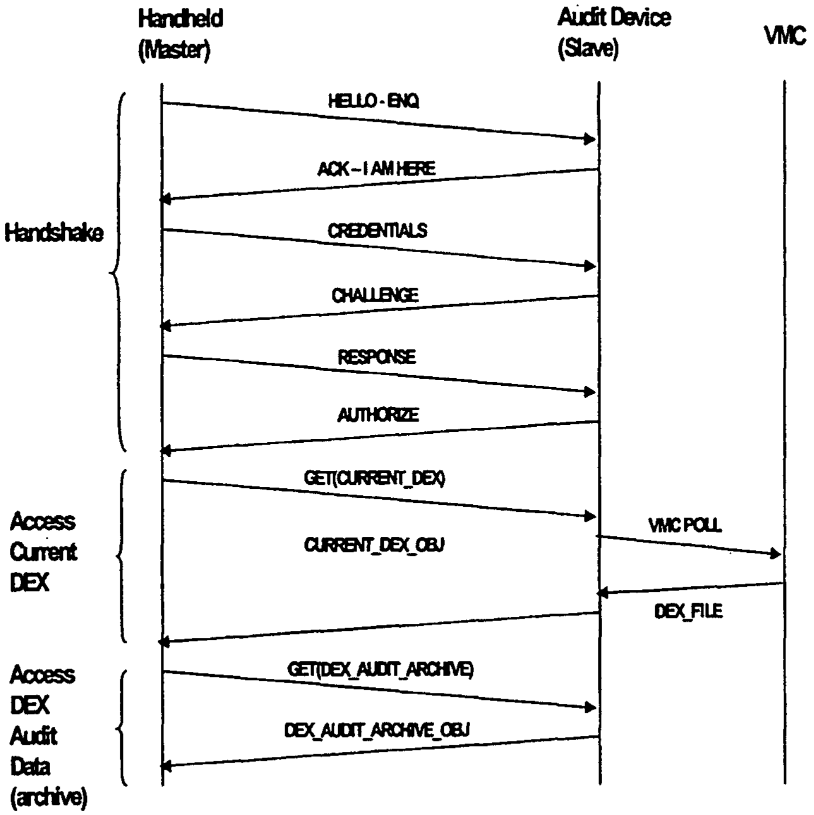


Figure 6.3.1-1: Sample Interaction Diagram

6.3.2 Communications API

In a Master-Slave implementation, the Audit Device exposes an API to the handheld in the form of a DLL or similar library. This library provides access to the various methods and properties (data) on the Audit Device. The library also implements the details of the communication and command protocol between the handheld and the Audit Device and abstracts it from the higher level application code on the handheld. An equivalent library is resident on the Audit Device firmware to perform the same abstraction function. The libraries also implement the required handshaking and authentication. A diagram illustrating the location of the library and DLL in the software/hardware stack is presented in **Figure 6.3.2-1**.

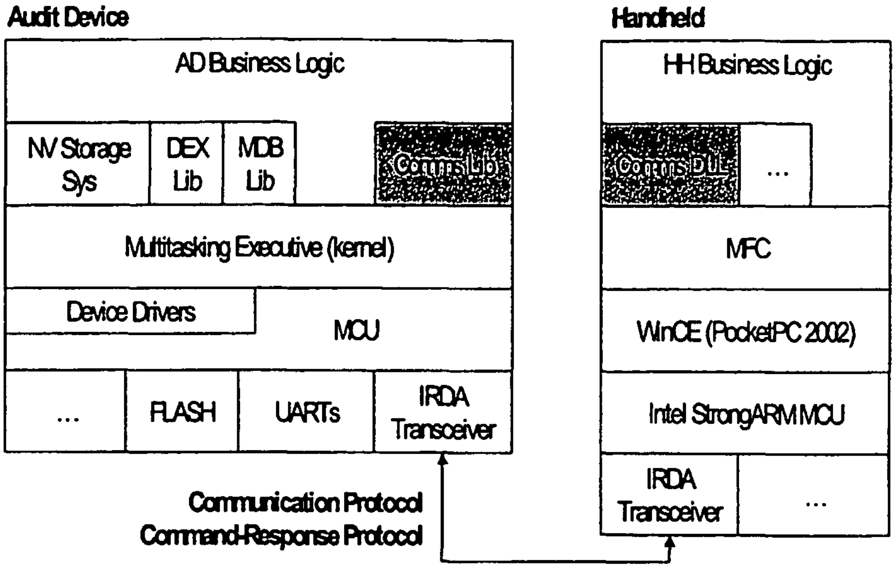


Figure 6.3.2-1: Hardware/Software Stack

VendCast Audit Device ("Glaux")
Provisional Patent Application

What is claimed is:

1. A system for auditing a vending machine, the system comprising:

an audit device mounted in a vending machine;

a multi-drop bus (MDB) interface in the audit device for communicating with an MDB interface of a vending machine controller (VMC) in the vending machine;

a DEX interface in the audit device for communicating with a DEX interface of a vending machine controller (VMC) in the vending machine;

a computer interface in the audit device for communicating with a hand held computer;

a clock in the audit device;

clock control logic in the audit device for automatically synchronizing the clock in the audit device with a clock in the handheld computer;

audit control logic in the audit device for automatically collecting DEX data and MDB data from the VMC;

nonvolatile memory in the audit device for storing the collected DEX data and MDB data;

the audit control logic storing timestamps with the DEX data and MDB data to record occurrence times for individual events and conditions within the vending machine;

authentication control logic for preventing unauthorized communications over the computer interface; and

transfer control logic for transferring the collected DEX data and MDB data from the audit device to the handheld computer, such that the collected DEX data and MDB data may be transferred from the audit device to a central operations center via the handheld computer.

2. A method for auditing a vending machine, the method comprising:

automatically collecting audit data in an audit device mounted in a vending machine according to predefined collection criteria ;

storing the audit data with timestamps to record occurrence times for individual events and conditions within the vending machine;

receiving authentication information from a handheld computer at the audit device;

in response to the authentication information, testing the authentication information for validity;

in response to receiving valid authentication data, synchronizing a clock in the audit device with a clock in the handheld computer and transferring at least some of the audit data to the handheld computer;

transmitting the audit data from the handheld computer to a host application on a central computer for analysis.